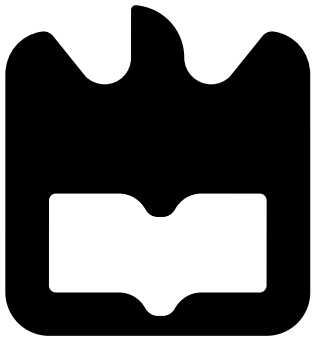
Universidade de Aveiro Departamento de Electrónica, Telecomunicações e Informática,
2020

Hugo Rafael Campinos Pintor

Utilização de dados estruturados na resposta a perguntas relacionadas com saúde
Using structured data to answer consumers health-related questions



Hugo Rafael Campinos Pintor

Utilização de dados estruturados na resposta a perguntas relacionadas com saúde
Using structured data to answer consumers health-related questions

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Informática, realizada sob a orientação científica de Dr. Sérgio Matos, Professor do Departamento Eletrónica Telecomunicações e Informática da Universidade de Aveiro

o júri / the jury

presidente / president

Professor Doutor Joaquim Arnaldo Carvalho Martins
Professor Catedrático da Universidade de Aveiro (por delegação do Reitor da Universidade de Aveiro

vogais / examiners committee

Professora Doutora Carla Alexandra Teixeira Lopes
Professora Auxiliar, Departamento de Engenharia Informática da Faculdade de Engenharia da Universidade do Porto

Professor Doutor Sérgio Guilherme Aleixo de Matos
Professor Auxiliar, Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro (orientador)

agradecimentos / acknowledgements

Aproveito para agradecer a todos os que me acompanharam e apoiaram, direta ou indiretamente, ao longo de todo o meu percurso académico.

De entre estes gostava de destacar de forma especial aqueles que me são mais próximos. Os meus pais por toda a paciência e apoio incondicional, em todos os momentos, e por terem feito de mim quem sou, pois apenas assim me foi possível ambicionar sempre mais. O meu irmão pelo seu espírito desafiante e provocador que muitas vezes foi a chave para me motivar nos momentos menos bons. Não esquecendo a restante família que apesar do seu menor envolvimento na minha vida académica, sei que foram pilares essenciais uma vez que apoiaram fortemente aqueles que mais se envolveram e aturaram ao longo deste capítulo.

Nunca esquecendo os amigos. Aqueles que precedem esta fase e que mesmo estando distantes sempre me apoiaram e aos que mais próximos estiveram e que levo para a vida, por tudo aquilo que viveram e partilharam comigo ao longo desta jornada.

Ao meu orientador pela disponibilidade, paciência e conhecimento partilhado. À Universidade de Aveiro pela formação e experiências proporcionadas.

E por último, mas não menos importante, um obrigado a ti por estares a ler esta tese.

palavras-chave sistema de resposta automático, saúde, pergunta

resumo Atualmente a forma mais comum de procurar informação é através da utilização de um motor de busca. Apesar de haver progresso os seus resultados continuam a ser maioritariamente baseados na devolução de uma lista de documentos onde estão presentes as palavras utilizadas na pesquisa, tendo o utilizador posteriormente que percorrer um conjunto dos documentos apresentados na esperança de obter a informação que procura. Para além de ser uma forma menos natural de procurar informação também é menos eficiente.

O objetivo para esta tese é melhorar esse processo de procura de informação, sendo neste caso o foco a área da saúde. Estas melhorias aconteceriam de duas formas diferentes, sendo a primeira a substituição da *query* normalmente utilizada em motores de busca, por algo que nos é mais natural - uma pergunta. E a segunda seria aproveitar a informação adicional a que temos acesso apenas no formato de pergunta, para fornecer os dados necessários à sua resposta em vez de uma lista de documentos onde um conjunto de palavras-chave estão presentes.

Sendo as redes sociais o local onde a busca por informação acontece através da utilização de perguntas, em substituição do que seria normal num motor de busca, pelo facto de a resposta nestas plataformas ser normalmente respondida por humanos e não máquinas. Parece assim ser o local natural para a recolha de perguntas para as quais temos o objetivo de fornecer uma ferramenta para a obtenção automática de uma resposta. O primeiro passo para ser possível fornecer esta resposta será a classificação das perguntas em diferentes tipos, tornando assim possível identificar qual a informação que se pretende obter. O segundo passo será identificar e categorizar as palavras de contexto biomédico presentes no texto fornecido, que seriam aquelas utilizadas caso a procura estivesse a ser feita utilizando as ferramentas convencionais. Tendo as palavras-chave sido identificadas e sabendo qual o tipo de informação que deverá estar presente na sua resposta. É agora possível mapear esta informação para um formato conhecido pelos computadores (*query*) e assim obter a informação pretendida.

keywords automatic question answering, biomedical question answering, user generated content

abstract

The current standard way of searching for information is through the usage of some kind of search engine. Even though there has been progress, it still is mainly based on the retrieval of a list of documents in which the words you searched for appear. Since the users goal is to find an answer to a question, having to look through multiple documents hoping that one of them have the information they are looking for is not very efficient.

The aim of this thesis is to improve that process of searching for information, in this case of medical knowledge in two different ways, the first one is replacing the usual keywords used in search engines for something that is more natural to humans, a question in its natural form. The second one is to make use of the additional information that is present in a question format to provide the user an answer for that same question instead of a list of documents where those keywords are present.

Since social media are the place where people replace the queries used on a search engine for questions that are usually answered by humans, it seems the natural place to look for the questions that we aim to provide with automatic answers. The first step to provide an answer to those questions will be to classify them in order to find what kind of information should be present in its answer. The second step is to identify the keywords that would be present if this was to be searched through the currently standard way. Having the keywords identified and knowing what kind of information the question aims to retrieve, it is now possible to map it into a query format and retrieve the information needed to provide an answer.

Contents

Contents i

List of Figures iii

List of Tables v

1 Introduction 1

2 State of the Art 4
2.1 Data acquisition . . . 4
2.1.1 Identifying useful questions . . . 4
2.1.2 Identifying "Interrogative Tweets" . . . 5
2.1.3 "Qweet" Extraction . . . 5
2.2 Question Classification . . . 7
2.2.1 Results . . . 7
2.3 Text Annotation . . . 10
2.3.1 TagTog . . . 10
Results . . . 10
2.3.2 PubTator . . . 12
Results . . . 12
2.3.3 Neji . . . 12
Results . . . 13
2.3.4 BERN . . . 14
Results . . . 14

3 Methodology 17
3.1 Components . . . 17
3.1.1 Data acquisition . . . 17
Twitter client . . . 17
Reddit client . . . 17
3.1.2 Question identification . . . 19
3.1.3 Question classification . . . 20
Training dataset . . . 20
3.1.4 Text annotation . . . 20
BERN . . . 21
3.1.5 Providing an answer . . . 21
3.2 Integration . . . 22

4 Results **25**
4.1 Twitter client . . . 25
4.2 Question classifier . . . 25
4.3 Application results . . . 26
4.3.1 Examples of manual analysis . . . 26
Example 1 . . . 26
Example 2 . . . 27
4.3.2 Results - using Reddit data . . . 29

5 Conclusions **35**
5.1 Future Work . . . 35

Bibliography **37**

Annex I **41**

List of Figures

1.1 Social networks number of users from 2004 to 2019. Available at: https://ourworldindata.org/rise-of-social-media. Accessed on 23 May, 2020 2

2.1 Entity recognition performance over all three corpora sizes [4]. 11
2.2 Unique entity recognition performance over all three corpora sizes [4]. 11
2.3 Comparison of precision, recall and F1-measure results achieved on AnEM and NCBI corpora for named entity recognition [2]. 13
2.4 BERN's [13] data flow diagram. 14

3.1 Example of a Reddit comment without context. 18
3.2 Post that Figure 3.1 was replying to. 19
3.3 Components interaction diagram. 23

4.1 Example 1 on Reddit website. 27
4.2 Example 2 on Reddit website. 28
4.3 BERN's annotation results for the corrections suggested. 32

List of Tables

2.1 Interrogative tweets identification results [20]. 5
2.2 Features used for qweet extraction [20]. 6
2.3 Qweet extraction results per feature combination [20]. 6
2.4 English tweets sample characteristics [20]. 7
2.5 Accuracy comparison of multiple machine learning models in question classification task [44]. 8
2.6 Comparison of state of the art approach of question classification in the medical field [30]. 9
2.7 Best classifier (CQT3) results per question type [30]. 9
2.8 Contest results for the tools used in Wei et al. [38]. 12
2.9 Performance comparison NER models for biomedical named entity recognition. For each entity type the highest scores are **bold** and the second highest are underlined[13]. 15

4.1 Question types distribution according to the used classifier. 29
4.2 Classification task results on real data. 29
4.3 Distribution of wrong classifications. 30
4.4 Annotation/Tagging results. 31
4.5 Causes of the missing tagged entities. 31
4.6 Successfully processed questions without an answer, reasons distribution. . . 33
4.7 Not answered questions due to not being mapped distribution per question type. 33

5.1 Classifiers comparison results. 41

Chapter 1

Introduction

The Internet has strongly spread its availability, specially in the past 20 years. Continuously growing as an important tool in our live in many different ways from what was its original purpose as a way to share knowledge.

Today, besides information, Internet also provides entertainment such as online games, streaming services and social networks which enable people from all over the world to connect.

Each media type serves multiple purposes. On one hand, besides being a way of entertainment, online games can also be used to connect with other people. On the other hand, streaming services can provide great scientific documentaries.

Alongside the evolution of the Internet's functions and uses, reading physical books as a way to gather knowledge has been getting increasingly obsolete as the Internet evolves in a globalized world as the one we live in, where everyone with an Internet connection can learn nearly anything.

Consumers have been increasingly relying on the Internet to supply answers for their health information needs [33]. A more concrete example is that among the 81% of American adults that use the Internet, 72% have searched for health information online [7].

Besides connecting people in a social way, social networks can also be a great tool to share and look for knowledge. A great percentage of people that use the Internet will probably have an account in one. Its number of users has never stopped growing since the start as shown in Figure 1.1. Knowing this, it seems counter-intuitive not to take advantage of these platforms to get our questions answered.

When answering questions regarding anyone's health, the reliability of the information provided should be a major concern. People with non specialized knowledge could provide wrong information which might cause damage to the person asking. It could also be that the person asking has some urgency in acquiring that information but for some reason doesn't have access to specialized professionals and have their questions answered, so the time that someone could be waiting for an answer might also be a concern.

Besides that, the process of creating an efficient query might be cognitively challenging for the average user [45], and in the end might even not be effective in retrieving relevant information [32, 43].

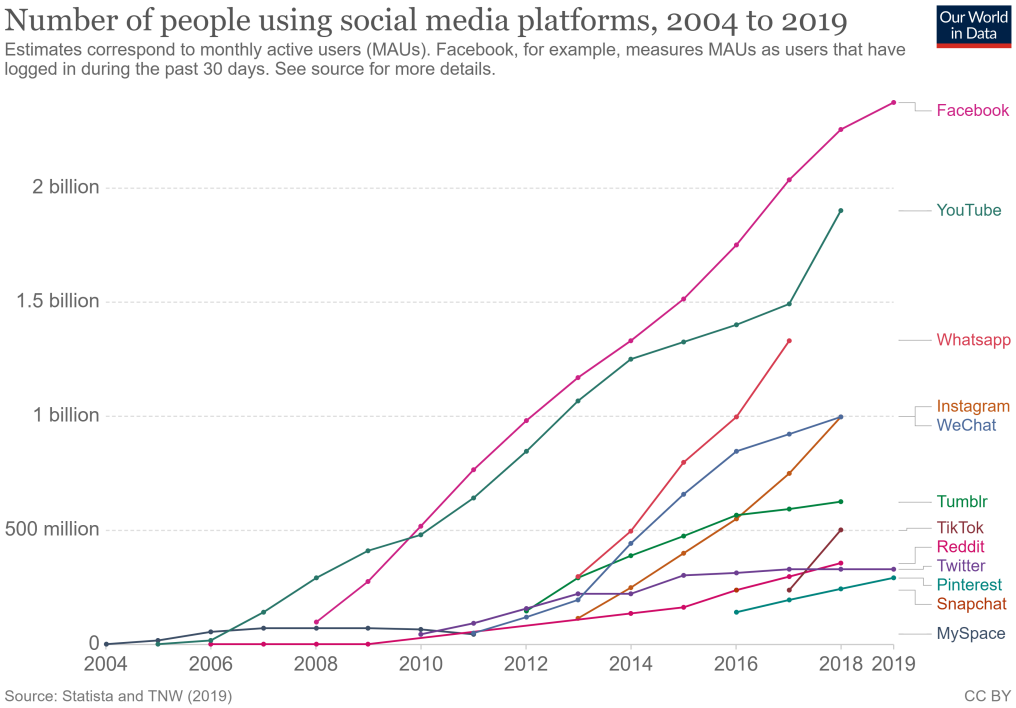


Figure 1.1: Social networks number of users from 2004 to 2019. Available at: https://ourworldindata.org/rise-of-social-media. Accessed on 23 May, 2020

What is being proposed here is to evaluate the possibility of combining the usage of structured data such as knowledge bases and social media to try to solve and/or minimize these problems. Using social media as a platform for people to expose their questions and use knowledge to provide the information needed to answer them.

This project can be summarized in six parts.

- Chose a social media to be used as a source of data .
- Identify and extract the questions found using NLP (Natural Language Processing) techniques.
- Classify all questions using machine learning classifiers.
- Annotate any health related terms present.
- Map the given question to some kind of query and execute it to retrieve the data needed to answer the given question.
- Provide an answer to the user.

The data acquisition should if possible happen using an API, and be able to reply the same way. Given the presence of some text characteristics that identify a question, it is then extracted and classified using a Support Vector Machine (SVM). BERN's [13] RESTful API is used to tag and classify the biomedical terms present in the question and its surrounding context. Knowing the question type, and the health related terms relevant to provide an answer, it is then mapped to a SPARQL query and Wikidata provides the data to create an answer.

Chapter 2 explores our three main technical challenges, question identification, health related questions classification and medical text annotation. Regarding the state-of-the-art techniques and tools developed to solve this problems.

Chapter 3 presents with more detail the implementation of each the components of this automatic question answering system and how they integrate each other. It includes the possible approaches for each part of the system and presents the decision making though process that led to the choice some approaches over others based on the presented research. The obtained results are then analysed in **Chapter 4**.

Chapter 5 presents the conclusions and suggests future work that can be done to improve what has been developed.

Chapter 2

State of the Art

The proposal for the present work is to try to use structured data such as knowledge bases to try to answer consumers medical questions in a given social media platform or complement other answers with additional information. For that objective to be possible there are some technical challenges that have to be solved. What is presented in this chapter is the current state of the art approaches to solving these challenges.

2.1 Data acquisition

For the development of a system that can identify and classify health related questions, a source for those will be a key component of the system. With that objective in mind and being that the focus of this thesis is to answer user generated questions, the usage of social media as a data source seems appropriate.

2.1.1 Identifying useful questions

Using social media as a data source provides a lot of data, which might be good for the intended purposes but that also represents a challenge. Not all this data will be useful and that being the case, it is crucial to find a way to identify useful content.

There are different types of content that incorporate questions, and not all of them should be processed and answered.

The analysis presented at Li et al. [20] is heavily focused on Twitter and for that matter the authors chose to use "Interrogative tweets" as a way to identify any tweet that contains a question and "qweet" refers to the present questions that were looking for an answer, help, explanation or information. It is important to know that from this point on the same terminology will be used.

Since all qweets are interrogative tweets, but an interrogative tweet is not always a qweet, we need a process that can separate them [20].

Some of the more common examples of interrogative tweets that are not qweets would be:

- Advertisement - Incorporating your business this year? Call us today for a free consultation with one of our attorneys. 855- 529-8753. http://buz.tw/FjJCV
- Question with answer - I even tried staying away from my Internet for a couple hours. The result? Insanity!

- Rhetorical question - You ruined my life and I'm supposed to like you?

2.1.2 Identifying "Interrogative Tweets"

There will always be false positives as well as false negatives, the precision and recall will most of the time have to be traded one for the other, and finding the best balance in that is essential. Since tweets are a rather small (280 characters maximum) informal kind of speech, that means that the usual/state of the art approaches might not be sufficient or the most adequate for the task. So Li et al. [20] experimented with some simple approaches, question mark (QM) presence, 5W1H ("what", "where", "when", "who", "how" and "why") combined with the usage of two context specific heuristics (H1, H2), where H1 means 5W1H must be the sentence starter and H2 looks for auxiliary verbs after (ex. "what" -> "what is"/"what are"), and the considered state of the art approach which uses sequential question patterns for identification [20] as well as combinations of all of the mentioned approaches.

Method	Results		
	Precision	Recall	F1
Question mark presence (QM)	0.969	0.846	0.903
QM and 5W1H	0.547	0.973	0.700
QM and refined 5W1H (Heuristic 1)	0.878	0.916	0.899
QM and refined 5W1H (Heuristic 2)	0.875	0.925	0.899
QM and refined 5W1H (Heuristic 1 & 2)	0.954	0.907	0.930
Miles Effron and Megan Winget [6]	0.960	0.855	0.904
Question Patterns (Confidence ≥ 0.7)	0.576	0.899	0.702
Question Patterns (Confidence ≥ 0.8)	0.715	0.872	0.786
Question Patterns (Confidence ≥ 0.9)	0.857	0.846	0.851

Table 2.1: Interrogative tweets identification results [20].

Miles Effron and Megan Winget [6] present an approach to question identification using the following set of rules:

- a question mark that is not part of a URL.
- the phrase I* [try*,like, need] to find.
- the phrase I* [try*, like, need] to know.
- the phrase I*m looking for .
- the phrase I* wonder*. In

2.1.3 "Qweet" Extraction

Tweets aside from other kinds of short text like Community Question Answering (CQA) questions and forums posts has some special characteristics, such as mentions, retweets and hashtags as well as links. That was also noted in Li et al. [20] that "Interrogative Tweets" could be split into two different parts, question and context. Even saying that context could possibly be more important for qweet identification than the question itself. Based on the

information above, four groups of qweet extraction features were developed, Question features (Q), Context features (C), Question-Context features (QC), Tweet-specific features (T) described in the image from the same article below.

Feature	Description
Question features (Q)	
Quoted question	Whether the question sentence is quoted from other sources
Strong feeling	Whether the question sentence contains strong feeling such as "???" and "?!"
Context features (C)	
URL	Whether the context contains any url
Phone number or Email	Whether the context contains any phone number or email
Strong feeling	Whether there is any strong feeling such as "!" follows the question sentence
Declarative sentence after a question	Whether there is any declarative sentence follows the question sentence
Word features	Unigram words appear in the contexts of tweets
Question-Context features (QC)	
Self ask self answer	Whether the tweet contains obvious self ask self answer pattern.
Question-url sameness	Whether the question is equal to a present url title.
Tweet-specific features (T)	
Username	Whether the tweet mentions other user's name.
Retweet	Whether the tweet is a Retweet.
Hashtag	Whether the tweet contains any hashtag.

Table 2.2: Features used for qweet extraction [20].

Based on these features, a Random Forrest classifier was used to predict whether the interrogative tweet was a qweet or not. These predictions were applied to various combinations of qweet extraction features which got the results in the table below.

Feature Set	Precision	Recall	Accuracy
Q	0.576	0.984	0.577
Q+QC+C	0.704	0.937	0.739
Q+QC+non-word C	0.714	0.906	0.73
Q+QC+non-word C+T	0.764	0.866	0.77
Q+QC+non-word C+Retweet	0.766	0.874	0.775
Q+QC+non-word C+@username	0.728	0.929	0.761
Q+QC+non-word C+Hashtag	0.702	0.89	0.721

Table 2.3: Qweet extraction results per feature combination [20].

The data set for testing the presented approach was a sample of one hour of tweets, retrieved from Twitter streaming API which could be our source of data. The sample was composed of 2045 English tweets, for which the table below describes its composition.

Number of tweets	2,045
Number of interrogative tweets	227
Number of qweets	127
Number of tweets containing URLs	442
Number of tweets containing hashtags	459
Number of Retweets	240
Number of tweets containing @	447
Number of tweets containing ?	222
Number of tweets containing 5W1H	293
Average number of words per tweet	12.48

Table 2.4: English tweets sample characteristics [20].

The results that were presented for both qweet extraction and interrogative tweet identification were based on the comparison of results between the automatic methods and two independent raters.

2.2 Question Classification

For question classification, if being manually constructed it implies a great amount of analysis work in order to figure out the multiple forms that each specific type of question has in order to achieve any significant amount of accuracy. In addition to that, the manual approaches are a lot less flexible than machine learning because any time you want to change the field of knowledge for which you are classifying questions you need to start a new question structure analysis and build a new classifier from scratch. When using a machine learning based approach to question classification if given a large enough dataset it will almost immediately achieve very decent results in any field of knowledge as shown by Dell Zhang and Wee Sun Lee [44] which compares Support Vector Machines (SVM) with four other machine learning algorithms Nearest Neighbors (NN), Naive Bayes (NB), Decision Tree (DT) and Sparse Network of Winnows (SNoW) using bag-of-words and bag-of-ngrams as the features.

2.2.1 Results

For the performance evaluation of the presented classifiers at Dell Zhang and Wee Sun Lee [44], the authors used a randomly selected subset of approximately 5500 labelled questions. For testing purposes a set of 500 labelled questions from TREC10 was used.

The achieved results were as follow:

Algorithm	Training set size				
	1000	2000	3000	4000	5500
NN	59.40%	64.60%	67.20%	67.40%	68.60%
NB	54.40%	58.40%	63.00%	65.00%	67.80%
DT	62.80%	72.20%	72.60%	73.00%	77.00%
SNoW	44.00%	67.00%	75.00%	55.80%	75.80%
SVM	65.00%	74.00%	74.80%	77.40%	79.20%

Table 2.5: Accuracy comparison of multiple machine learning models in question classification task [44].

Knowing what works in a generalized manner is relevant, but since this work is focusing on biomedical questions done by consumers it is also important to analyse what has been done in this more specific field of knowledge. Consumer medical related questions will most of the time focus on a disease that the person as been diagnosed with, what diseases may be the cause for their symptoms, what is the prognosis and possible treatments [30]. For that matter is important to correctly classify the question to be able to identify what information is needed to provide an answer. The definition of question types is also an important step and Roberts et al. [30] suggests a set of 13 types, based on the type of answer:

- Anatomy - relation of some part of the body with a given disease.
- Cause - tries to identify the cause of a given disease or symptom.
- Complication - looking for consequences of a disease, without specifying signs or symptoms.
- Diagnosis - questions targeted at retrieving information to help with a diagnosis.
- Information - open ended question, probably could be answered with a combination of the above types.
- Management - looking for treatment or cure information.
- Manifestation - trying o identify signs or symptoms of a disease.
- OtherEffect - side effects of a given disease or symptom.
- PersonOrg - looking for an entity, either person or organization to provide more information or treatment.
- Prognosis - what the future will look like for someone with a given disease, life expectancy, impacts on daily life.
- Susceptibility - regarding to information on the distribution and spread of a given disease.
- Other - medical questions that do not fit any of the above types.
- NotDisease - questions that are not supported by the QA system.

Besides the definition of question types, the set of features to use in the classification is also a prevalent factor. For this Roberts et al. [30] did an analysis and comparison with other state-of-the-art approaches. From the analysed set, the ones that could be considered more relevant for the comparison are the works by Li and Roth [21], because of being the first presented machine learning method for question classification based on the answer type, and by Yu and Cao [42], because of the similar question types to their own. After that analysis they proposed three different sets of features CQT(1-3) that are combinations of features from the analysed work, for which the comparison results are presented in Table 2.6. The classifier used was SVM.

Approach	Accuracy
Bag-of-words	76.89%
Li and Roth [21]	77.45%
Yu and Cao [42]	78.43%
Liu et al. [22]	76.37%
Patrick and Li [28] - UQC	77.41%
Patrick and Li [28] - QTC	77.76%
Patrick and Li [28] - GTC	77.76%
CQT1	79.01%
CQT2	80.40%
CQT3	82.42%

Table 2.6: Comparison of state of the art approach of question classification in the medical field [30].

Question Type	#Annotations	Precision	Recall	F1
Anatomy	12	66.7	16.7	26.7
Cause	119	83	78.2	80.5
Complication	32	65.4	53.1	58.6
Diagnosis	229	83.1	75.1	78.9
Information	520	86.3	93.7	89.9
Management	673	91.4	89.7	90.6
Manifestation	103	87.3	86.4	86.8
NotDisease	16	20	6.2	9.5
OtherEffect	275	64.7	66.5	65.6
Other	38	63.2	31.6	42. 1
PersonOrg	128	87. 1	78.9	82.8
Prognosis	313	78.9	79.9	79.4
Susceptibility	420	78	86	81.8

Table 2.7: Best classifier (CQT3) results per question type [30].

The training dataset was composed of 1467 publicly available requests from Genetic and Rare Diseases Information Center (GARD). Each request frequently had more than a single question which was separated in different entries, resulting in a set of 2937 questions.

2.3 Text Annotation

To be able to retrieve the data necessary to answer any kind of question, we need to know what are the main terms present. For that reason, an annotation system that focuses on the same field of knowledge (biomedical) as the system being developed is necessary. It will not only allow for the identification of the most relevant terms in the question and its context but will also classify them, which will be useful in later stages of development.

2.3.1 TagTog

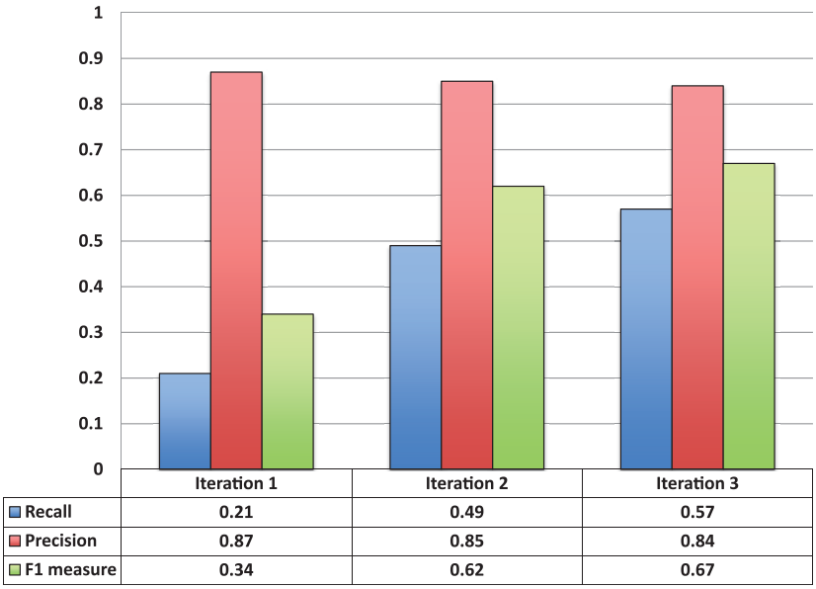
TagTog allows for the usage of machine learning models like the ones provided out of the box such as a general-purpose named entity recognizer (NER) using conditional random fields (CRFs), which are probabilistic models used to label sequence data and have multiple advantages over other approaches to this problem, such as hidden Markov models and stochastic grammars [15] . These CRFs were trained with common features used in previous systems.

The authors point out that instead of using the approach that achieves the best possible results, performance is traded for a slight increase in speed by using a sole CRF model instead of taking advantage of AIIAGMT[4] which uses an additional backwards parsing model that greatly improve the achieved results [11] . That is justified by tagtog team with the increased usability of the user-interactive system [4].

Results

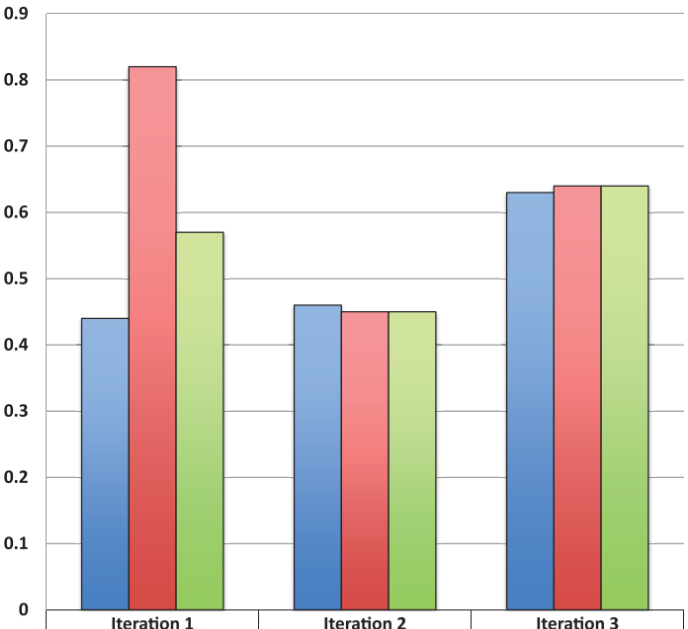
As for benchmarking, tagtog used the FlyBase corpus with some standard named entity recognition (NER) evaluation measures such as "precision (P), recall (R) and F1 measure (F1)" [4]. Three iterations were made on the tagtog annotation model with different training and test sets.

There were 3 different test iterations. The third one uses the most complete training data and is tested against what the authors call the Gold Standard. It can be said that this iteration results are the most relevant. Where the analyzed tool achieved 57% recall, 84% precision and 67% F1 in entity recognition, and 63% recall, 64% precision and 64% F1 in unique entity recognition.



	Iteration 1	Iteration 2	Iteration 3
Recall	0.21	0.49	0.57
Precision	0.87	0.85	0.84
F1 measure	0.34	0.62	0.67

Figure 2.1: Entity recognition performance over all three corpora sizes [4].



	Iteration 1	Iteration 2	Iteration 3
Recall	0.44	0.46	0.63
Precision	0.82	0.45	0.64
F1 measure	0.57	0.45	0.64

Figure 2.2: Unique entity recognition performance over all three corpora sizes [4].

2.3.2 PubTator

PubTator annotation system is a result of combining multiple tools that had already been extensively tested in multiple text-mining competitions. Tools such as GenTUKit [12] for gene identification, GenNorm [36] for gene normalization, SR4GN [37] for species identification used to aid bio-curators working with specific organisms and support gene normalization, DNorm [16] for diseases, tmVar [35] for mutations and a dictionary based approached for chemicals.

For what it is intended to be developed those annotation models that focus on diseases, mutations and chemicals are the most relevant since those will probably be the most commonly mentioned in social media.

It is also worth pointing out that besides the annotation service that was just described, PubTator also provides a search function that retrieves PubMed documents based on the tagged terms, which might be useful in the later phases of development.

As for the annotation function, at the time of the publication there were support for three annotation tasks, document triage, entity annotation and relationship annotation.

Results

As for presented results there aren't any PubTator specific values for the task, because even though they provide a web service for annotation purposes, the main focus of PubTator was the usability of the system, and since it makes use of previously tested tools, it is most relevant to our work the results achieved by those tools in their corresponding evaluations.

Bio-entity	Text-mining tool	F1 Score
Gene (mention)	GeneTUKit	82.97%
Gene (normalization)	GenNorm	92.89%
Disease	DNorm	80.90%
Species	SR4GN	85.42%
Chemical	A dictionary-based lookup approach	53.82%
Mutation	tmVar	93.98%

Table 2.8: Contest results for the tools used in Wei et al. [38].

2.3.3 Neji

Neji is an annotation tool built around four fundamental aspects, modularity, scalability, speed and usability. It integrates several state of the art tools for biomedical entity recognition [25] in a variety of different evaluation corpus with entity types that include disorders, anatomy, chemicals and others achieving F-scores 85, 82 and 87 percent respectively and using both dictionary and machine learning based approaches which have their respective advantages over the others depending on the situation. If working with precise and well defined vocabularies like diseases and species although other difficulties such as the need of creating a unique resource with all the identifiers that are usually spread across multiple data sources might be a concern, usually the usage of a dictionary based solution is recommended. On the other hand with highly variable and dynamic vocabularies such as gene and protein names, machine learning based solutions are the standard approach but not without additional challenges such as the need for results normalization.

For that matter the processing pipeline developed has 5 different steps.

- To interpret and filter the input data.
- Pre-processing with the objective of simplifying the entity recognition process.
- Entity identification.
- Post-processing to improve the results and solve some problems resultant from the recognition process.
- Output the result of all the previous steps in a structured matter.

For each of the aforementioned steps, multiple methods were used to be able to adapt to different contexts [2].

Results

It was tested against three different Gold Standard corpora such AnEM (Anatomical entity mention detection) [27] and NCBI disease corpus, which are the most relevant because of covering the terms that seem to be most commonly used by the non medical experts which will be our source of data, for which the results achieved are presented at Figure 2.3. These cover the biomedical type of concepts which CRAFT [1] lacks at. But since it is the largest and the one with the most biomedical concept coverage it still was the main driver for NEJI's development and improvement.

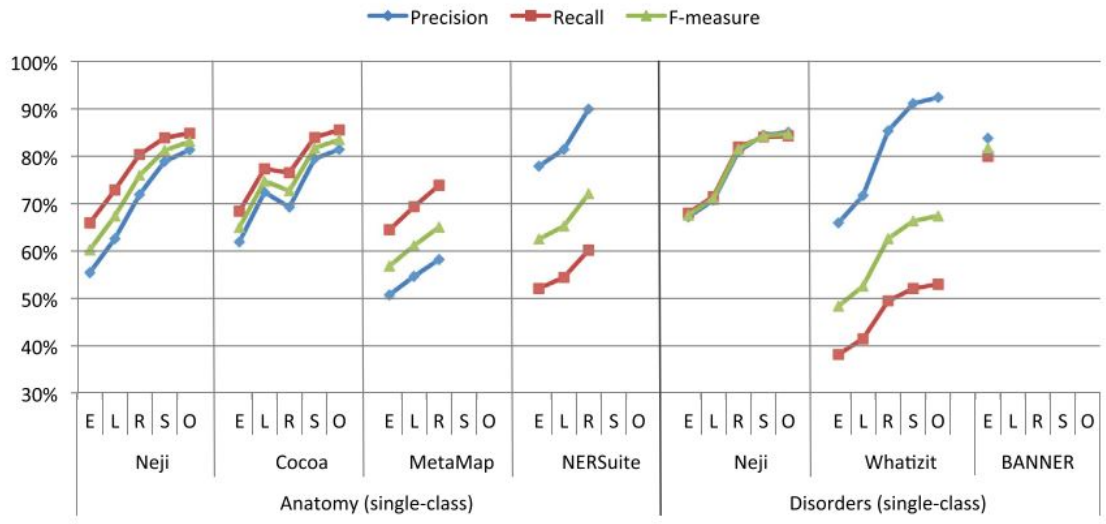


Figure 2.3: Comparison of precision, recall and F1-measure results achieved on AnEM and NCBI corpora for named entity recognition [2].

As shown NEJI achieves similar results to other state of the art tools when it comes to anatomy entity tagging tasks and largely surpasses its peers in disorders.

Besides its annotation capabilities using the available methods at the time of development, it also offers great capacity for improvement since it allows for an easy way to integrate new dictionaries and machine-learning models, enabling the platform to use the best annotation tools available.

2.3.4 BERN

BERN (Biomedical named Entity Recognition and multi-type Normalization) as the ones presented above is a biomedical text-mining tool, but with an improved set of features.

The author of the tool state that the first web-based tool with the ability to discover new entities using BioBERT NER [19].

It also incorporates a set of probability based decision rules to identify overlapping entities. The example given by the authors is "The androgen is synthesized from...", androgen can either be classified as a gene/protein or drug/chemical depending on the context.

In a simplified manner, the way data flows through the system for what might be our use case, where the input is raw text, is as follows.

tmVar 2.0 [40] receives the raw text and tags the mutations present. It forwards the result to BioBERT NER [19] to get genes/proteins, diseases, drugs/chemicals and species tagged. After that it uses its probability-based decision rules to identify overlapping entities. Then the normalization process is run for each of the entity types and the result is returned.

To facilitate the understanding of the process, the visual representation of it is shown in Figure 2.4.

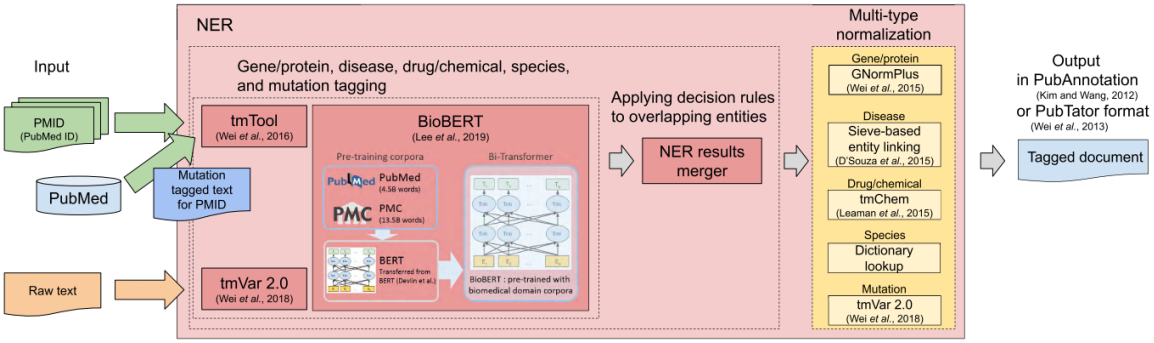


Figure 2.4: BERN's [13] data flow diagram.

Results

The table that follows presents the comparison results of their proposed solution (BERN) [13] with other state of the art systems used in the same NER tasks and using the same dataset for each of the entity types being tagged.

Text mining tools	Entity types	Pre-trained NER models	Test sets	Precision	Recall	F1-score
BERN	Gene/Protein	BioBERT [19]	BC2GM [26]	**0.8516**	**0.8365**	**0.8440**
-		Sachan et al. [31]		0.8181	0.8157	0.8169
-		MTM-CW of Wang et al. [34]		0.8210	0.7942	0.8074
-		CollaboNet [41]		0.8049	0.7899	0.7973
ezTag, tmTool, PubTerm		GNormPlus [39]		0.7840	0.7920	0.7880
-		Giorgi and Bader [9]		0.7862	0.7871	0.7866
-		LSTM-CRF (iii) of Habibi et al. [41]		0.7750	0.7813	0.7782
BERN	Disease	BioBERT [19]	NCBI disease [5]	0.8904	**0.8969**	**0.8936**
-		Sachan et al. [31]		0.8641	0.8831	0.8734
-		MTM-CW of Wang et al. [34]		0.8586	0.8642	0.8614
-		CollaboNet [41]		0.8548	0.8727	0.8636
-		Giorgi and Bader [9]		0.8262	0.8695	0.8472
-		LSTM-CRF (iii) of Habibi et al. [41]		0.8531	0.8358	0.8444
-		D3NER [10]		0.8503	0.8380	0.8441
-		Lou et al. [23]		**0.9072**	0.7489	0.8205
ezTag		aTaggerOne [17]		0.8510	0.8080	0.8290
ezTag		bTaggerOne [17]		0.8350	0.7960	0.8150
tmTool, PubTerm		DNorm [16]		0.8030	0.7630	0.7820
BERN	Drug/Chemical	BioBERT [19]	BC4CHEMD [14]	0.9223	**0.9061**	**0.9141**
-		MTM-CW of Wang et al. [34]		0.9130	0.8753	0.8937
-		CollaboNet [41]		0.9078	0.8701	0.8885
-		Giorgi and Bader [9]		0.8343	0.8883	0.8605
-		LSTM-CRF (iii) of Habibi et al. [41]		0.8783	0.8545	0.8662
-		Att-BiLSTM-CRF of Luo et al. [24]		**0.9229**	0.9001	0.9114
tmTool, PubTerm		tmChem (Model 2) [18]		0.8909	0.8575	0.8739
BERN	Species	BioBERT [19]	LINNAEUS [8]	0.9384	0.8611	0.8981
-		Giorgi and Bader [9]		0.9280	0.9429	0.9354
-		LSTM-CRF (iii) of Habibi et al. [41]		0.9357	0.9324	0.9340
-		LINNAEUS [8]		**0.9710**	**0.9430**	**0.9570**
ezTag, tmTool, PubTerm		SR4GN [37]		0.8582	0.8528	0.8555
BERN, ezTag	Mutation	tmVar 2.0 [40]	tmVar2 [40]	0.9725	0.9040	0.9370
tmTool, PubTerm		tmVar [35]	MutationFinder [3]	0.9880	0.8962	0.9398

Table 2.9: Performance comparison NER models for biomedical named entity recognition. For each entity type the highest scores are **bold** and the second highest are underlined [13].

As seen above BERN achieves the best results of all the tested systems for the three most relevant entity types, Gene/Protein, Disease and Drug/Chemical.

Chapter 3

Methodology

This chapter presents the different implementations for each of the developed components as well as how they are integrated with each other.

3.1 Components

This section will serve the purpose of explaining the implementation of each of the system components with more detail.

3.1.1 Data acquisition

For the development of an automatic question answering system, a source for questions to be answered will be a key component. For that purpose and being that the focus of the project is user generated content, the usage of social media as a data source seems adequate, for which the key features are abundance of questions, easy way to retrieve its content and capacity to reply from an external application.

Twitter client

The following component was developed in python, taking advantage of an already existing wrapper for both Twitter streaming and search API called Tweepy. To use the streaming API, the client needs to provide a list of search terms. Given that the size of that list is limited to approximately 400 terms for the free version of the service, the first step was to define which terms would be used. For that purpose the twitter search API was used to rank all the diseases available at the "The Genetic and Rare Diseases Information Center (GARD)" website (https://rarediseases.info.nih.gov/glossary) by the number of occurrences. The top 400 of those results were then used as the search parameters for the streaming API

Reddit client

Reddit was found to also be a suitable platform for searching user generated health related questions.

The Reddit client was also developed in python taking advantage of an already existing library, PRAW: The Python Reddit API Wrapper.

Given the nature/structure of the chosen platform it was possible to verify the presence of the kind of questions this system aims to answer before starting the development. In that search it was found that some suitable subreddits for that search could be "medical" and "AskDocs".

The information needed to make use of the aforementioned wrapper is only the name of a subreddit and a number of submissions to be retrieved. Even though it was limited to a single subreddit per execution, given the limit of 30 requests per minute there is no problem in iterating through the starting small list of subreddits. Even if the number of subreddits grow over the 30 that can be searched per minute, it would still not be a problem since there is no need to search all subreddits every minute and if a need for it was found a second instance of the client using a different account would easily solve the problem.

Questions were identified and extracted using the methods presented at Section 3.1.2. As for the definition of what would be the context for the identified question there were three different iterations.

As a first approach ignoring all comments and only looking for the questions present at the submission. This meant an heavy limitation on the number of answers that could be provided, since most of the interactions, discussion and questions are usually present at the comment section.

As of the second iteration the comments started being analysed, but as isolated entities, this caused problems because of questions similar to the example shown at Figure 3.1. It wouldn't be possible to answer because there is no presence of either a symptom or disease in the comment.

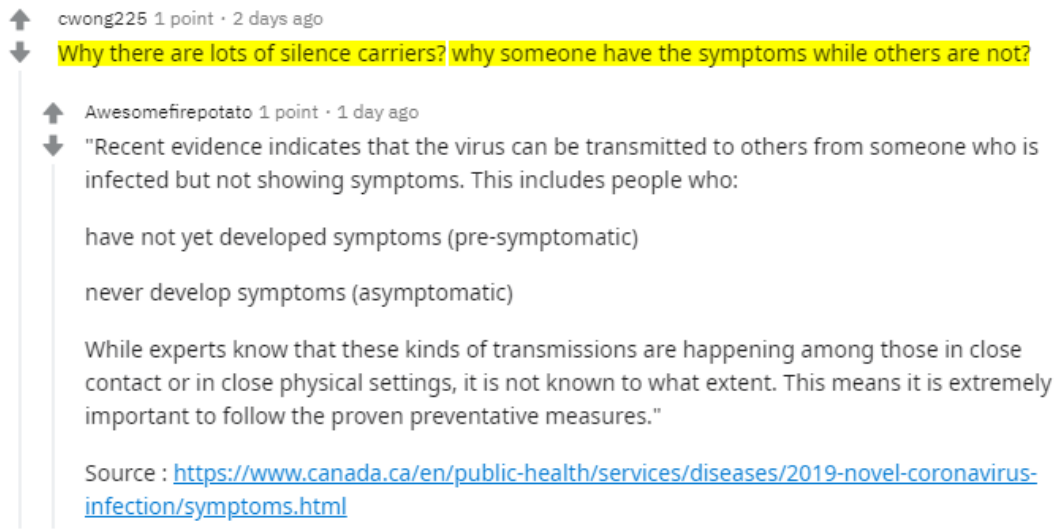


Figure 3.1: Example of a Reddit comment without context.

As a third iteration, comments from the same branch (chain of replies) and the submission content and title started being considered as context. With this came the solution for a large percentage of the questions being unanswered.

Implementation below:

```
def process_submissions(self):
  for subreddit_name in self.subreddits_list:
    subreddit = self.reddit.subreddit(subreddit_name)
    hot_submissions = subreddit.hot(limit=self.submissions_num)
    for submission in hot_submissions:
      self.current_submission_url = submission.url
      if not submission.stickied:
      submission_text = self.text_cleaning(submission.title + " " + submission.
    selftext)
      self.save_questions(text=submission_text, context=submission_text)
      submission.comments.replace_more(limit=None)
      for root_comment in submission.comments:
        try:
        self.process_comments(comment=root_comment, context=submission_text)
        except AttributeError:
        print("Log: Found comment with no body") # deleted comments

def process_comments(self, comment, context):
  comment_text = self.text_cleaning(comment.body)
  context = context + " ; " + comment_text
  self.save_questions(text=comment_text, context=context)
  for reply in comment.replies:
    self.process_comments(reply, context)

def save_questions(self, text, context):
  for question in text.split("?")[:-1]:
    is_question, text, indexes = question_identification(question+"?")
    if is_question:
      self.questions.append({"context": context, "question": text[indexes[0]:indexes
    [1]], "submission_url": self.current_submission_url})
      self.output_file.write(json.dumps({"context": context, "question": text[indexes
    [0]:indexes[1]],
        "submission_url": self.current_submission_url}) + "\n")
```

Listing 3.1: Reddit submissions and comments processing.

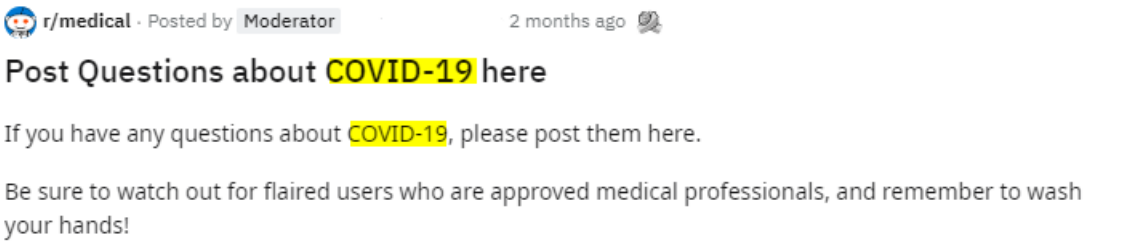


Figure 3.2: Post that Figure 3.1 was replying to.

With the additional information from Figure 3.2 it would be possible to answer the question.

3.1.2 Question identification

A method to identify the questions to be answered was developed with the objective of simplifying manual data analysis and the development of other components. The approach used on this identification process was the same as the one presented by Li et al. [20] as

"QM and refined 5W1H (Heuristic 1 & 2)" which meant that the questions always start with 5W1H followed by an auxiliary verb and that after both of those question parts there still is a question mark.

The developed question identification method would only find a single question in a given text even if multiple were present. For that reason the Reddit client splits the content by "?" before submitting for evaluation. As shown in Code Snippet 3.1.

```
def question_identification(text):
  original_text = text.strip()
  text = re.sub(' +', ' ', text.strip().lower())
  question_prefixes = ('what', 'where', 'when', 'who', 'how', 'why')
  aux_verbs = ( 'do', 'does', 'am', 'is', 'are', 'was', 'were', 'have', 'has', 'had',
    'can', 'could', 'may', 'might', 'shall', 'should', 'will', 'would', 'ought to' )
  prefix_index = -1
  selected_prefix = ''
  is_qweet = False

  for prefix in question_prefixes:
    if prefix in text:
      if text.find(prefix) > prefix_index and "?" in text[text.find(prefix):]:
        prefix_index = text.find(prefix)
        selected_prefix = prefix
  if prefix_index != -1:
    for aux_verb in aux_verbs:
      if aux_verb in text:
        if text.find(aux_verb) == (prefix_index + len(selected_prefix) + 1):
          is_qweet = True
          break

  return is_qweet, original_text, (prefix_index, prefix_index + text[prefix_index:].
    find('?') + 1)
```

Listing 3.2: Question identification and validation process.

3.1.3 Question classification

The choice of using SVM as a question classifier with a bag-of-words as the features obtained using a TFIDF (term frequency–inverse document frequency) vectorizer without any preprocessing or using any stop words dictionary is justified at Section 4.2. The achieved results could be improved by using the custom set of features from the best performance present at Section 2.6. The reason why a custom set of features was defined as a low priority improvement was due to the high effort and potentially low reward.

Training dataset

The classification model training dataset used for both the final implementation and the classification models tests present at Table 5.1 was composed of 2937 questions, with 13 different question types manually annotated as described in Roberts et al. [29].

3.1.4 Text annotation

As for the text annotation, four different possibilities were evaluated. Those were tagtog, PubTator, NEJI and BERN. Tagtog being the one to show worse results and given the fact

that its main purpose is to have a great user-interactive system meant at one point trading the best possible results for a slight increase in speed and that being the case it was the first one to be disregarded as an option. When comparing PubTator, NEJI and BERN, looking at raw results on diseases which is the main entity type we're looking to tag terms at, both NEJI and PubTator show very similar results, NEJI would have been the chosen one because of being able to distinguish diseases and symptoms which might be useful in later processing phases. But both of these options show results that are approximately 10% inferior to what BERN achieved meaning that this will be the annotation tool of choice.

BERN

As for integration of BERN in the system we used python requests library to interact with the provided API. It allowed for the identification of genes, drugs, species, diseases and symptoms (both classified as diseases) which would enable for the retrieval of the information needed to answer five (Cause, Complication, Diagnosis, Information and OtherEffect) out of the thirteen types of questions.

3.1.5 Providing an answer

After retrieving and identifying a question. It is then classified and the whole context is tagged. After these steps what is left is to gather the information needed. The way we do it is through searching the tagged entities and their relevant relations, for the question type it has been classified with, in Wikidata using a SPARQL query.

Given the presence of a disease and/or symptom in the list of tagged entities it is then possible to identify the needed information for its answer.

```
instances_of_diseases_query = """SELECT DISTINCT ?disease ?diseaseLabel
  WHERE {
    ?disease wdt:P31/wdt:P279* wd:Q12136
    SERVICE wikibase:label { bd:serviceParam wikibase:language "en". }
  }"""

instances_of_symptoms_query = """SELECT DISTINCT ?symptom ?symptomLabel
  WHERE {
    ?symptom wdt:P31/wdt:P279* wd:Q169872
    SERVICE wikibase:label { bd:serviceParam wikibase:language "en". }
  }"""

instances_of_clinical_sign_query = """SELECT DISTINCT ?symptom ?symptomLabel
  WHERE {
    ?symptom wdt:P31/wdt:P279 * wd:Q1441305
    SERVICE wikibase:label { bd:serviceParam wikibase:language "en". }
  }"""
```

Listing 3.3: SPARQL queries for preloading all diseases, symptoms and clinical signs

Since Wikidata does not allow for the usage of the entity names directly in its queries, first we load all the entities that are instances of disease, symptom or clinical sign to our application. Then it is possible to translate the tagged terms into their correspondent identifiers to evaluate their relevant relations.

3.2 Integration

The integration of all the system components happen has follows.

A set of subreddits and a number of submissions to look into per subreddit is provided to the Reddit client. It iterates through the submissions and all its retrieved comments looking for questions. Every time a question is found it saves the whole comment/submission and all preceding context (other comments it replies to and the original submission). The question is classified. And the whole comment/submission is annotated using BERN web service. After that we should have the necessary data to map the question using the SPARQL Mapper and retrieve the information needed to produce an answer to the given type of question.

```
bern_client = BernClient()
wikidata_client = WikidataClient()
file = open("./resources/GARD_qdecomp.final.qtd.txt", "r").readlines()
du = DataUtils(vectorizer="tfidf", include_stopwords=False, basic_nlp=False,
    ngram_range=(1, 1))

reddit_client = RedditClient(submissions_num=500)
reddit_client.process_submissions()
questions = reddit_client.questions
y, x = du.transform_data(file)
classifier = sk_models.get_svm_classifier(x, y)

for question in questions:
  print("Submission URL: " + question["submission_url"])
  print("Question: " + question["question"])
  print("Full Context: " + question["context"])
  question["class"] = du.le.inverse_transform(classifier.predict(du.vect.transform([
    question["question"]])))[0]
  print("Question class: " + question["class"])
  question["annotations"] = bern_client.annotate_and_translate(question["context"])
  print("Resumed Annotation results: " + str(question["annotations"]))
  if question["class"] in ["Cause", "Complication", "Diagnosis", "OtherEffect"]:
    print("Generated answers: ")
    for disease in question["annotations"]["disease"]:
    wikidata_client.answer_diseases_questions(disease)
    for symptom in question["annotations"]["disease"]:
    wikidata_client.answer_symptoms_questions(symptom)
```

Listing 3.4: Question answering system main function.

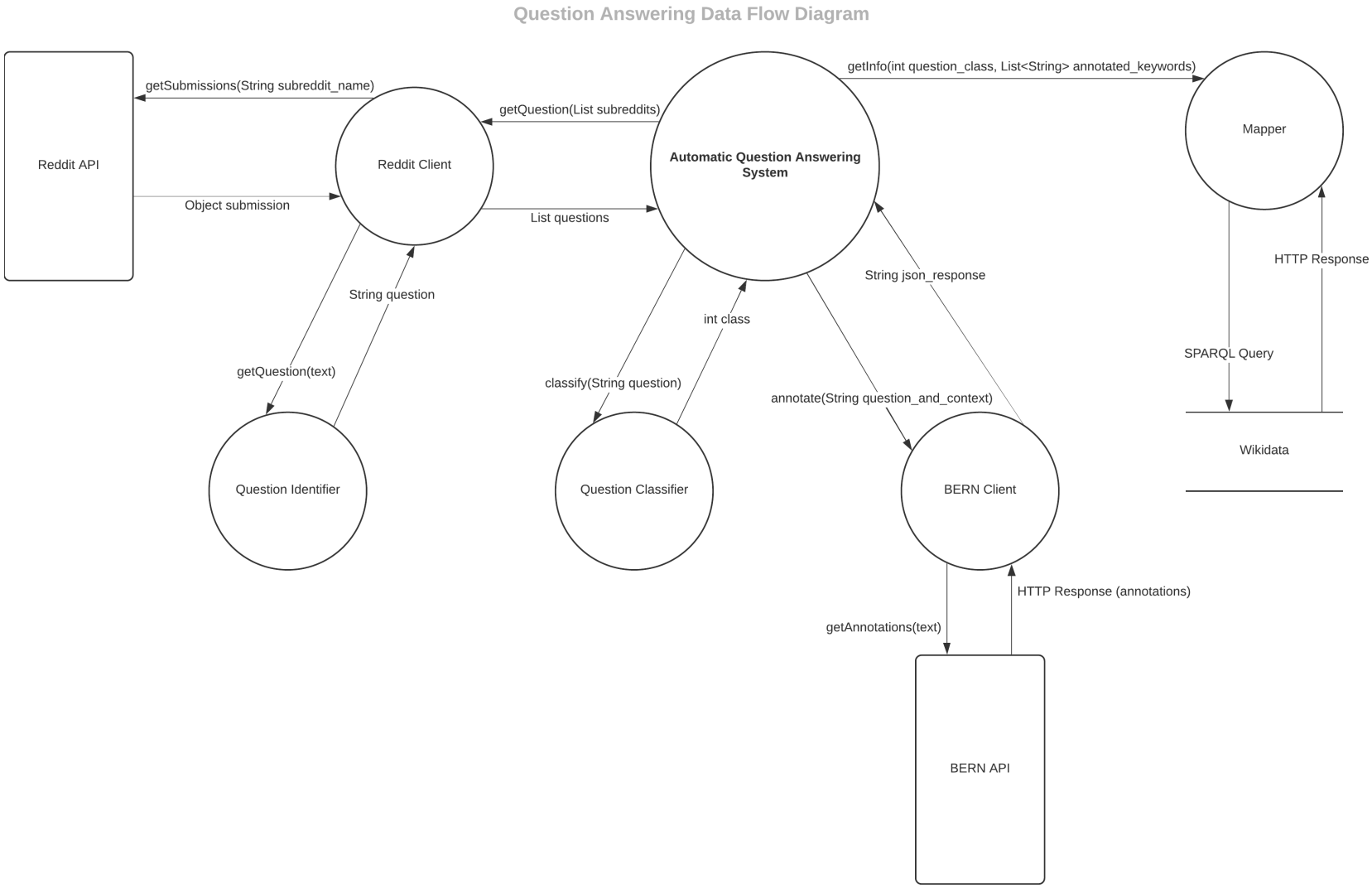


Figure 3.3: Components interaction diagram.

Chapter 4

Results

The following chapter is going to present some of the performance values for the system components as isolated entities. And later as a complete application.

4.1 Twitter client

After the development of the twitter client and question extraction process, the client was running for a week retrieving and storing Interrogative Tweets. After a manual analysis on the acquired data it was concluded that most of the Interrogative Tweets would fall in one of two categories, question followed by the answer or advertising, making them not suitable for testing the developed automatic question answering system capabilities.

4.2 Question classifier

Given the nature of the data being classified and the limited amount of questions to train the SVM classifier. Other alternatives such as Multilayer Perceptron(MLP) and Random Forrest (RF) were put to the test against what could be considered the state of the art approach given the results shown at Section 2.2.

The comparison used multiple combinations of configuration parameters to achieve the best possible results. Those parameters were the usage of a stop-words dictionary (True or False), the usage of some basic preprocessing (True or False), the range of ngrams to be used by the vectorizer (All the possible combinations from (1,1) - bag of words to (3,3) a bag of ngrams that considers sequences of 3 words) and the type of vectorizer to be used (Count or TFIDF). The test consisted of 100 runs for each of the aforementioned classification models and all possible combinations of the configuration parameters.

From the results present at Table 5.1, it was confirmed that the best classifier for the given training dataset would be SVM with an average of 76.94% correct predictions without preprocessing or the usage of any stop words dictionary, using a TFIDF Vectorizer and a bag of words (Ngram range of (1,1)).

The closest contender was MLP with 76.93% of successful predictions using a Count Vectorizer and being all other configuration parameters equal.

4.3 Application results

Given the nature of the developed system and its data source, an automatic evaluation method that comprises the whole system is something that would be very difficult to achieve. The results and examples that follow were achieved through manual evaluation.

4.3.1 Examples of manual analysis

Here are some illustrative examples of the manual evaluation process for each question being considered in **Subsection** 4.3.2.

Example 1

Submission URL: `https://www.reddit.com/r/medical/comments/gxxne6/risk_of_lymphedema_when_removing_lymph_node_from/`

Question: "What are the risks associated with such a procedure?"

Full Context: "Risk of Lymphedema when removing lymph node from neck. [19M] Hello, my GP suggested to have one of my lymph nodes in the neck removed but I am afraid of lymphedema (buildup of lymph liquid). Are my fears justified? What are the risks associated with such a procedure? Thank you very much for any advice."

Question Class: Management

BERN's response:


```
{
  "project": "BERN",
  "sourcedb": "",
  "sourceid": "71deced7f86adf28f1e78e8155afa18f63231d772af177e8779df062",
  "text": "Risk of Lymphedema when removing lymph node from neck. [19M] Hello, my GP suggested to have one of my
       lymph nodes in the neck removed but I am afraid of lymphedema (buildup of lymph liquid). Are my fears
       justified? What are the risks associated with such a procedure? Thank you very much for any advice.",
  "denotations": [{
    "id": ["OMIM:153100", "MESH:D008209", "BERN:254155101"],
    "span": {
      "begin": 8,
      "end": 18
    },
    "obj": "disease"
  }, {
    "id": ["OMIM:153100", "MESH:D008209", "BERN:254155101"],
    "span": {
      "begin": 153,
      "end": 163
    },
    "obj": "disease"
  }],
  "timestamp": "Sun Jun 07 20:27:40 +0000 2020",
  "logits": {
    "disease": [
      [{
        "start": 8,
        "end": 18,
        "id": "OMIM:153100\tMESH:D008209\tBERN:254155101"
      }, 0.9999998807907104],
      [{
          "start": 153,
          "end": 163,
          "id": "OMIM:153100\tMESH:D008209\tBERN:254155101"
      }, 0.9999998807907104]
    ],
    "gene": [],
    "drug": [],
    "species": []
  }
}
```


Annotation results:


```
{'disease': ['Lymphedema'], 'gene': [], 'drug': [], 'species': []}}
```

Answers: None

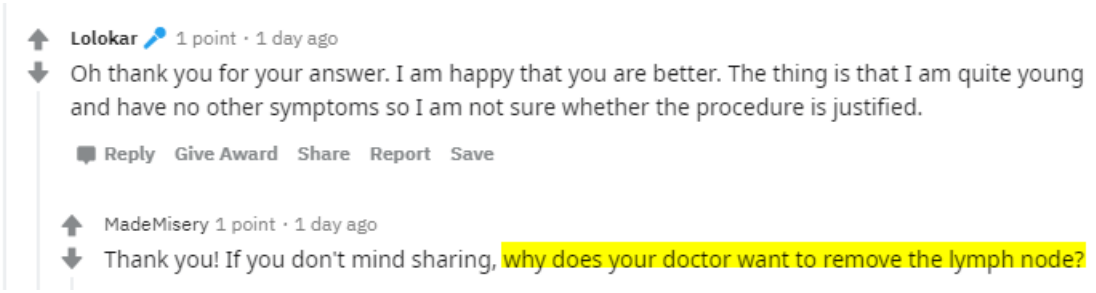


Figure 4.1: Example 1 on Reddit website.

Analysis: Question was well classified. BERN's successfully tagged all the diseases. An answer could possibly be provided, but a mapping function for the given question type is not yet implemented.

Example 2

Submission URL: `https://www.reddit.com/r/medical/comments/gxxne6/risk_of_lymphedema_when_removing_lymph_node_from/`

Question: "why does your doctor want to remove the lymph node?"

Full Context: "Risk of Lymphedema when removing lymph node from neck. [19M] Hello, my GP suggested to have one of my lymph nodes in the neck removed but I am afraid of lymphedema (buildup of lymph liquid). Are my fears justified? What are the risks associated with such a procedure? Thank you very much for any advice. ; Your fears are justified, but just having one removed shouldn't be that bad. I had thyroid cancer that spread to a lymph node in my neck. They removed between 60 to 70 lymph nodes (most of them were tiny) just to be safe and make sure the cancer didn't spread further. After I healed I did notice some slight lymphedema in my face and under my chin (gave me a nice embarrassing double chin for a bit). I went to a massage therapist that did lymphatic massage. I only saw her about 10 times and the lymphedema got a lot better. Also don't use antiperspirants as they can clog the lymph nodes under your arms and make drainage more difficult and the lymphedema more pronounced. Most brands for men have some that are just deodorants. Old Spice is a good one. ; Oh thank you for your answer. I am happy that you are better. The thing is that I am quite young and have no other symptoms so I am not sure whether the procedure is justified. ; Thank you! If you don't mind sharing, why does your doctor want to remove the lymph node?"

Question Class: OtherEffect

BERN's reponse:


```
{
  "project": "BERN",
  "sourcedb": "",
  "sourceid": "71deced7f86adf28f1e78e8155afa18f63231d772af177e8779df062",
  "text": "Risk of Lymphedema when removing lymph node from neck. [19M] Hello, my GP suggested to have one of my
       lymph nodes in the neck removed but I am afraid of lymphedema (buildup of lymph liquid). Are my fears
       justified? What are the risks associated with such a procedure? Thank you very much for any advice.",
  "denotations": [{
      "id": ["OMIM:153100", "MESH:D008209", "BERN:254155101"],
      "span": {
        "begin": 8,
        "end": 18
```

```
      },
      "obj": "disease"
    }, {
      "id": ["OMIM:153100", "MESH:D008209", "BERN:254155101"],
      "span": {
        "begin": 153,
        "end": 163
       },
      "obj": "disease"
    }],
  "timestamp": "Sun Jun 07 20:27:40 +0000 2020",
  "logits": {
    "disease": [
      [{
        "start": 8,
        "end": 18,
        "id": "OMIM:153100\tMESH:D008209\tBERN:254155101"
        }, 0.9999998807907104],
      [{
        "start": 153,
        "end": 163,
        "id": "OMIM:153100\tMESH:D008209\tBERN:254155101"
        }, 0.9999998807907104]
      ],
    "gene": [],
    "drug": [],
    "species": []
  }
}
```


Annotation results:


```
{'disease': ['Lymphedema', 'thyroid cancer', 'cancer'], 'gene': [], '
    drug': [], 'species': []}
```


Answers: Effects of Lymphedema are: ['Stemmer sign']
Causes of Lymphedema are: []
Effects of thyroid cancer are: []
Causes of thyroid cancer are: ['multiple endocrine neoplasia type 2A', 'ionizing radiation', 'iodine-131']
Effects of cancer are: ['cancer pain']
Causes of cancer are: ['somatic mutation']

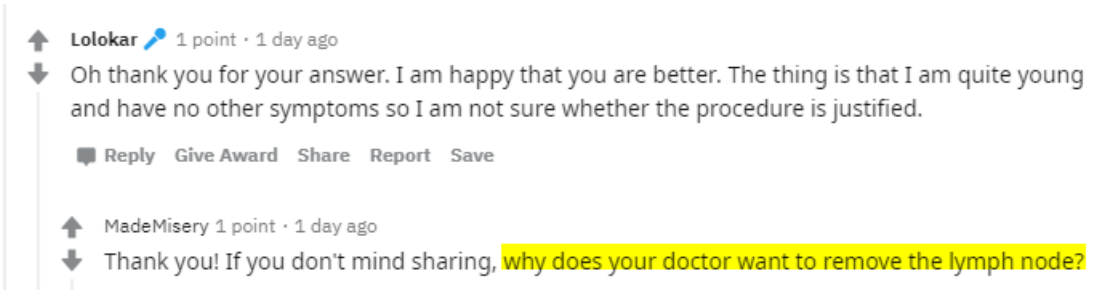


Figure 4.2: Example 2 on Reddit website.

Analysis: Question wrongly classified, should be classified as "Other" since it is a medical question but doesn't fit any of the other 11 biomedical categories. BERN's successfully tagged all the diseases. If it actually was an "OtherEffect" type of question the necessary information was probably present in the provided answers.

4.3.2 Results - using Reddit data

The data set used is from the Top 500 most popular submissions for the 2 subreddits used, which resulted in the extraction of 272 questions.

Type	Quantity	Percentage
Anatomy	0	0.00%
Cause	11	4.04%
Complication	0	0.00%
Diagnosis	7	2.57%
Information	73	26.84%
Management	91	33.46%
Manifestation	4	1.47%
OtherEffect	41	15.07%
PersonOrg	8	2.94%
Prognosis	21	7.72%
Susceptibility	16	5.88%
Other	0	0.00%
NotDisease	0	0.00%

Table 4.1: Question types distribution according to the used classifier.

From this dataset, the ones that went through the whole pipeline and tried to generate an answer were 132 (48.53%)

The following data in this chapter is the result of manual analysis on the 272 questions dataset.
Below is the result of the question classification task.

Result	Quantity
Correctly classified	145
Wrongly classified	122
Wrongly extracted	5

Table 4.2: Classification task results on real data.

From those wrongly classified the error was distributed as follows.

Classifier result	Manual classification	Quantity
Information	Diagnosis	24
Information	Cause	4
Information	Other	4
Information	Management	1
Management	Diagnosis	6
Management	Information	3
Management	Cause	7
OtherEffect	Other	6
OtherEffect	Cause	2
OtherEffect	Management	2
OtherEffect	Information	1
OtherEffect	Anatomy	3
OtherEffect	NonMedical	7
OtherEffect	Diagnosis	4
PersonOrg	NotDisease	1
Prognosis	Information	2
Prognosis	Other	2
Prognosis	Diagnosis	5
Susceptibility	Information	2
Susceptibility	NotDisease	2
Susceptibility	Diagnosis	2
Cause	Management	1
Diagnosis	Other	1
Diagnosis	NotDisease	1
Diagnosis	Management	2
Information	NotDisease	7
Information	PersonOrg	1
Management	NotDisease	3
Management	PersonOrg	1
Management	Other	3
OtherEffect	Prognosis	1
PersonOrg	Management	1
Prognosis	NotDisease	2
Prognosis	Management	3
Susceptibility	Cause	2
Susceptibility	Other	2

Table 4.3: Distribution of wrong classifications.

As for the annotation/tagging task the results are as shown below.

Result	Quantity
Correctly annotated	124
Missing information	130
NotDisease	18

Table 4.4: Annotation/Tagging results.

From the ones that have missing information the reasons distribution is:

Annotation missing information cause	Quantity
No medical terms	58
Failed to tag entities	43
Misspelling of the entities	1
Out of medical spectrum	1
No medical terms - Image	26
Case specific question	1

Table 4.5: Causes of the missing tagged entities.

Most of the occurrences of missing medical terms that could be tagged happened mainly because of 3 reasons.

Fifty eight of those times were caused by the informal nature of the platform. Where possibly, due to the lack of knowledge, most people chose to describe instead of using the term that could replace the given description and be successfully tagged.

Example:
Submission URL: `https://i.redd.it/f9z359rlhe351.jpg`
Question: what is this?
Full Context: i felt my throat hurting since yesterday, but it got worse today. what is this? what can i do to cure it? and should i seek medical treatment or will it go away?
Question class: Information
BERN's reponse:


```
{
  "project": "BERN",
  "sourcedb": "",
  "sourceid": "13f5d3eddbc715730133ceaf79048c92567588a9729549b0da6e0446",
  "text": "i felt my throat hurting since yesterday, but it got worse today. what is this? what can i do to cure it
      ? and should i seek medical treatment or will it go away?",
  "denotations": [],
  "timestamp": "Sun Jun 07 22:59:38 +0000 2020",
  "logits": {
    "disease": [],
    "gene": [],
    "drug": [],
    "species": []
  }
}
```


Annotation results:


```
{'disease': [], 'gene': [], 'drug': [], 'species': []}
```


Generated answers: None

Analysis:If instead of using "i felt my throat hurting" was used "throat soreness" or "throat pain" to describe the symptom. BERN's would have been able to tag the entity and it would

be possible to search on Wikidata for possible causes. As shown below.

Figure 4.3: BERN's annotation results for the corrections suggested.

Twenty six times there were no entities to tag due to the question being dependent on an image analysis.
Forty three other times, BERN failed to tag some entities.

Example:
Submission URL:
`https://www.reddit.com/r/medical/comments/gy4e8k/how_does_reducing_your_bacterial_load_let_your/`
Question: How does reducing your bacterial load let your immune system maintain a low enough level of bacteria so as to stop acne?
Full Context: How does reducing your bacterial load let your immune system maintain a low enough level of bacteria so as to stop acne? Doesn't [D.Tan's last para.](https://medicalsciences.stackexchange
>Personal thoughts: Antibiotics most likely never completely eradicate a bacterial population, but by bringing down the bacterial load, you enable your immune system to maintain the remaining bacteria at a low enough level to avoid clinical symptoms. contradict [Dr Graham Chiu's answer](https://medicalsciences.stackexchange.com/a/5605): Commensal bacteria on the skin are not controlled by the immune system ? To wit, how does reducing the bacterial load enable your immune system to maintain a low enough level of remaining bacteria to avoid clinical symptoms, if your immune system can't control or kill *C. acnes*?
Question class: Management
Annotation results:

```
{'disease': ['acne'], 'gene': [], 'drug': [], 'species': ['C. acnes']}
```

Analysis: There were two terms that should be tagged. Those are "bacterial load" and "immune system". These words are part of the biomedical field but are not covered by any of the categories present at BERN's models.

There were 32 questions for which both the classification and the annotation parts were correct, but only 2 of the answers of those could be considered successful. However it should be mentioned that 7 other questions could have been answered if correctly classified.
From that set of 32 questions, the reasons why 30 of them were not answered are distributed as show at Table 4.6

Reason	Quantity
Not mapped	19
No info available	11

Table 4.6: Successfully processed questions without an answer, reasons distribution.

From those that were not mapped, the distribution per question type is presented below.

Question type	Quantity
Management	15
PersonOrg	1
Prognosis	2
Susceptibility	1

Table 4.7: Not answered questions due to not being mapped distribution per question type.

Example:
Submission URL:
`https://www.reddit.com/r/medical/comments/gy4e8k/how_does_reducing_your_bacterial_load_let_your/`
Question: How does reducing your bacterial load let your immune system maintain a low enough level of bacteria so as to stop acne? **Full Context**: How does reducing your bacterial load let your immune system maintain a low enough level of bacteria so as to stop acne? Doesn't [D.Tan's last para.](https://medicalsciences.stackexchange.com/a/23293) >Personal thoughts: Antibiotics most likely never completely eradicate a bacterial population, but by bringing down the bacterial load, you enable your immune system to maintain the remaining bacteria at a low enough level to avoid clinical symptoms. contradict [Dr Graham Chiu's answer](https://medicalsciences.stackexchange.com/a/5605): Commensal bacteria on the skin are not controlled by the immune system ? To wit, how does reducing the bacterial load enable your immune system to maintain a low enough level of remaining bacteria to avoid clinical symptoms, if your immune system can't control or kill *C. acnes*? **Question class**: Management

Annotations results:

```
{'disease': ['acne'], 'gene': [], 'drug': [], 'species': ['C. acnes']}
```

Analysis: There were two terms that should be tagged. Those are "bacterial load" and "immune system". These words are part of the biomedical field but are not covered by any of the categories present at BERN's models.

Chapter 5

Conclusions

This thesis was developed with the intent of helping consumers have their biomedical questions answered without having to go through the cognitive challenging process of transforming a question into a query. And providing immediate, useful and trustful information.

The way it proposed to do so, was through the use of structured data to retrieve the necessary information based on the classification of the question and tagging of the present biomedical entities.

These objectives were achieved to some degree. The information that the developed system provides is trustworthy. It has the ability to answer a question immediately if it is correctly classified, the necessary biomedical terms are present and Wikidata has them.

Here is where the biggest problem lies, Wikidata is an open domain knowledge base. Which means that much of the field specific terms are not available or don't have a well defined set of relations with the necessary entities.

The next big problem is the nature of the platform that was used to evaluate its performance, usually Reddit users have a very casual way of posing their questions, being very descriptive instead of using well defined terms. While this is very natural of an interaction in social media, the developed system doesn't have the capabilities to work with descriptions in place of biomedical terms. If it was to be tested in a platform dedicated to answering users medical questions. It could perform better because it would be a less casual interaction, the person would have to explicitly think of visiting that platform to have their questions answered. And would possibly have a better though and structured question.

The question classification process could use some improvement, but it is not detrimental to the system's performance because there are types of questions that look for very similar information. It can also be minimized using additional logic based on the types of tagged entities.

BERN is very good in tagging the types of entities it aims to. A minor limitation it has is that it doesn't cover all of the entity types needed. For example "bacterial load" and "immune system" can't be placed in any of the provided types.

5.1 Future Work

For future work the recommendations are to improve on the previously mentioned flaws:

- Replace Wikidata with a field specific knowledge base.

- Map the remaining question types into SPARQL queries.
- Add the capability of translating descriptions into terms and/or change the platform used for one dedicated to consumer biomedical question answering.
- Use an additional NER system for the entities not covered by BERN.
- Improve the question classifier, by replacing the automatic features used with the ones suggested in Roberts et al. [30] and/or by improving the dataset.

Bibliography

[1] Michael Bada, Miriam Eckert, Donald Evans, Kristin Garcia, Krista Shipley, Dmitry Sitnikov, William A. Baumgartner, K. B. Cohen, Karin Verspoor, Judith A. Blake, and Lawrence E. Hunter. Concept annotation in the CRAFT corpus. *BMC Bioinformatics*, 13(1):161, jul 2012.

[2] David Campos, Sérgio Matos, and José L. Oliveira. A modular framework for biomedical concept recognition. *BMC Bioinformatics*, 14(1), sep 2013.

[3] J Gregory Caporaso, William A Baumgartner, David A Randolph, K Bretonnel Cohen, and Lawrence Hunter. MutationFinder: a high-performance system for extracting point mutation mentions from text. *Bioinformatics*, 23(14):1862–1865, jul 2007.

[4] Juan Miguel Cejuela, Peter McQuilton, Laura Ponting, Steven J Marygold, Raymund Stefancsik, Gillian H Millburn, and Burkhard Rost. tagtog: interactive and text-mining-assisted annotation of gene mentions in PLOS full-text articles. *Database*, 2014:bau033–bau033, apr 2014.

[5] Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. NCBI Disease Corpus: A Resource for Disease Name Recognition and Concept Normalization. *J Biomed Inform*, 47:1–10, 2014.

[6] Miles Efron and Megan Winget. Questions are content: A taxonomy of questions in a microblogging environment. *Proceedings of the American Society for Information Science and Technology*, 47(1):1–10, nov 2010.

[7] Duggan M. Fox S. Pew Internet & American Life Project, a Project of the Pew Research Center. 2013.

[8] Martin Gerner, Goran Nenadic, and Casey M. Bergman. LINNAEUS: A species name identification system for biomedical literature. *BMC Bioinformatics*, 11(1):85, feb 2010.

[9] John M Giorgi and Gary D Bader. Transfer learning for biomedical named entity recognition with neural networks. *Bioinformatics*, 34(23):4087–4094, dec 2018.

[10] Thanh Hai Dang, Hoang-Quynh Le, Trang M Nguyen, and Sinh T Vu. D3NER: biomedical named entity recognition using CRF-biLSTM improved with fine-tuned embeddings of various linguistic information. *Bioinformatics*, 34(20):3539–3546, 2018.

[11] C.-N. Hsu, Y.-M. Chang, C.-J. Kuo, Y.-S. Lin, H.-S. Huang, and I-F. Chung. Integrating high dimensional bi-directional parsing models for gene mention tagging. *Bioinformatics*, 24(13):i286–i294, jul 2008.

[12] Minlie Huang, Jingchen Liu, and Xiaoyan Zhu. GeneTUKit: a software for document-level gene normalization. *Bioinformatics (Oxford, England)*, 27(7):1032–3, apr 2011.

[13] Donghyeon Kim, Jinhyuk Lee, Chan Ho So, Hwisang Jeon, Minbyul Jeong, Yonghwa Choi, Wonjin Yoon, Mujeen Sung, and Jaewoo Kang. A Neural Named Entity Recognition and Multi-Type Normalization Tool for Biomedical Text Mining. *IEEE Access*, 7:73729–73740, 2019.

[14] Martin Krallinger, Obdulia Rabal, Florian Leitner, Miguel Vazquez, David Salgado, Zhiyong Lu, Robert Leaman, Yanan Lu, Donghong Ji, Daniel M Lowe, Roger A Sayle, Riza Theresa Batista-Navarro, Rafal Rak, Torsten Huber, Tim Rocktäschel, Sérgio Matos, David Campos, Buzhou Tang, Hua Xu, Tsendsuren Munkhdalai, Keun Ho Ryu, SV Ramanan, Senthil Nathan, Slavko Žitnik, Marko Bajec, Lutz Weber, Matthias Irmer, Saber A Akhondi, Jan A Kors, Shuo Xu, Xin An, Utpal Kumar Sikdar, Asif Ekbal, Masaharu Yoshioka, Thaer M Dieb, Miji Choi, Karin Verspoor, Madian Khabsa, C Lee Giles, Hongfang Liu, Komandur Elayavilli Ravikumar, Andre Lamurias, Francisco M Couto, Hong-Jie Dai, Richard Tzong-Han Tsai, Caglar Ata, Tolga Can, Anabel Usié, Rui Alves, Isabel Segura-Bedmar, Paloma Martínez, Julen Oyarzabal, and Alfonso Valencia. The CHEMDNER corpus of chemicals and drugs and its annotation principles. *Journal of Cheminformatics*, 7(S1):S2, dec 2015.

[15] John Lafferty, Andrew McCallum, and Fernando C N Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML '01 Proceedings of the Eighteenth International Conference on Machine Learning*, 2001.

[16] Robert Leaman, Rezarta Islamaj Dogan, and Zhiyong Lu. DNorm: disease name normalization with pairwise learning to rank. *Bioinformatics (Oxford, England)*, 29(22):2909–17, nov 2013.

[17] Robert Leaman and Zhiyong Lu. TaggerOne: joint named entity recognition and normalization with semi-Markov Models. *Bioinformatics*, 32(18):2839–2846, sep 2016.

[18] Robert Leaman, Chih Hsuan Wei, and Zhiyong Lu. TmChem: A high performance approach for chemical named entity recognition and normalization. *Journal of Cheminformatics*, 7(Suppl 1 Text mining for chemistry and the CHEMDNER track), 2015.

[19] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, sep 2019.

[20] Baichuan Li, Xiance Si, Michael R. Lyu, Irwin King, and Edward Y. Chang. Question identification on twitter. In *Proceedings of the 20th ACM international conference on Information and knowledge management - CIKM '11*, page 2477, New York, New York, USA, 2011. ACM Press.

[21] Xin Li and Dan Roth. Learning question classifiers. In *Proceedings of the 19th international conference on Computational linguistics* -, volume 1, pages 1–7, Morristown, NJ, USA, 2002. Association for Computational Linguistics.

[22] Feifan Liu, Lamont D. Antieau, and Hong Yu. Toward automated consumer question answering: Automatically separating consumer questions from professional questions in the healthcare domain. *Journal of Biomedical Informatics*, 44(6):1032–1038, dec 2011.

[23] Yinxia Lou, Yue Zhang, Tao Qian, Fei Li, Shufeng Xiong, and Donghong Ji. A transition-based joint model for disease named entity recognition and normalization. *Bioinformatics*, 33(15):2363–2371, aug 2017.

[24] Ling Luo, Zhihao Yang, Pei Yang, Yin Zhang, Lei Wang, Hongfei Lin, and Jian Wang. An attention-based BiLSTM-CRF approach to document-level chemical named entity recognition. *Bioinformatics*, 34(8):1381–1388, apr 2018.

[25] Sérgio Matos. Configurable web-services for biomedical document annotation. *Journal of Cheminformatics*, 10(1):68, dec 2018.

[26] Alexander A Morgan, Zhiyong Lu, Xinglong Wang, Aaron M Cohen, Juliane Fluck, Patrick Ruch, Anna Divoli, Katrin Fundel, Robert Leaman, Jörg Hakenberg, Chengjie Sun, Heng-hui Liu, Rafael Torres, Michael Krauthammer, William W Lau, Hongfang Liu, Chun-Nan Hsu, Martijn Schuemie, K Bretonnel Cohen, and Lynette Hirschman. Overview of BioCreative II gene normalization. *Genome Biology*, 9(Suppl 2):S3, 2008.

[27] Tomoko Ohta, Sampo Pyysalo, Jun ' Ichi Tsujii, and Sophia Ananiadou. Open-domain Anatomical Entity Mention Detection. Technical report, 2012.

[28] Jon Patrick and Min Li. An ontology for clinical questions about the contents of patient notes. *Journal of Biomedical Informatics*, 45(2):292–306, apr 2012.

[29] K Roberts, K Masterton, M Fiszman, H Kilicoglu, and D Demner-Fushman. Annotating Question Types for Consumer Health Questions. *Lrec 2014 - Ninth International Conference on Language Resources and Evaluation*, 2014.

[30] Kirk Roberts, Halil Kilicoglu, Marcelo Fiszman, and Dina Demner-Fushman. Automatically classifying question types for consumer health questions. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2014:1018–27, 2014.

[31] Devendra Singh Sachan, Pengtao Xie, Mrinmaya Sachan, and Eric P Xing. Effective Use of Bidirectional Language Modeling for Transfer Learning in Biomedical Named Entity Recognition. Technical report, 2018.

[32] Amanda Spink, Yin Yang, Jim Jansen, Pirrko Nykanen, Daniel P. Lorence, Seda Ozmutlu, and H. Cenk Ozmutlu. A study of medical and health queries to web search engines. *Health information and libraries journal*, 21(1):44–51, 2004.

[33] Nupur Tustin. The role of patient satisfaction in online health information seeking. *Journal of Health Communication*, 15(1):3–17, jan 2010.

[34] Xuan Wang, Yu Zhang, Xiang Ren, Yuhao Zhang, Marinka Zitnik, Jingbo Shang, Curtis Langlotz, and Jiawei Han. Cross-type Biomedical Named Entity Recognition with Deep Multi-Task Learning. *Bioinformatics*, 2015.

[35] C.-H. Wei, Bethany R. Harris, H.-Y. Kao, and Zhiyong Lu. tmVar: a text mining approach for extracting sequence variants in biomedical literature. *Bioinformatics*, 29(11):1433–1439, jun 2013.

[36] Chih-Hsuan Wei and Hung-Yu Kao. Cross-species gene normalization by species inference. *BMC Bioinformatics*, 12(S8):S5, dec 2011.

[37] Chih-Hsuan Wei, Hung-Yu Kao, and Zhiyong Lu. SR4GN: A Species Recognition Software Tool for Gene Normalization. *PLoS ONE*, 7(6):e38460, jun 2012.

[38] Chih-Hsuan Wei, Hung-Yu Kao, and Zhiyong Lu. PubTator: a web-based text mining tool for assisting biocuration. *Nucleic Acids Research*, 41(W1):W518–W522, jul 2013.

[39] Chih-Hsuan Wei, Hung-Yu Kao, and Zhiyong Lu. GNormPlus: An Integrative Approach for Tagging Genes, Gene Families, and Protein Domains. *BioMed Research International*, 2015:1–7, 2015.

[40] Chih-Hsuan Wei, Lon Phan, Juliana Feltz, Rama Maiti, Tim Hefferon, and Zhiyong Lu. tmVar 2.0: integrating genomic variant information from literature with dbSNP and ClinVar for precision medicine. *Bioinformatics*, 34:80–87, 2018.

[41] Wonjin Yoon, Chan Ho So, Jinhyuk Lee, and Jaewoo Kang. CollaboNet: collaboration of deep neural networks for biomedical named entity recognition. *BMC Bioinformatics*, 20(S10):249, may 2019.

[42] Hong Yu and Yong-Gang Cao. Automatically extracting information needs from Ad Hoc clinical questions. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2008:96–100, nov 2008.

[43] Qing T. Zeng, Jonathan Crowell, Robert M. Plovnick, Eunjung Kim, Long Ngo, and Emily Dibble. Assisting consumer health information retrieval with query recommendations. *Journal of the American Medical Informatics Association*, 13(1):80–90, jan 2006.

[44] Dell Zhang and Wee Sun Lee. Question classification using support vector machines. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval - SIGIR '03*, page 26, New York, New York, USA, 2003. ACM Press.

[45] Yan Zhang. Contextualizing consumer health information searching: An analysis of questions in a social Q&A community. In *IHI'10 - Proceedings of the 1st ACM International Health Informatics Symposium*, pages 210–219, New York, New York, USA, 2010. ACM Press.

Annex I

Configurations				Classifiers					
				SVM		RF		MLP	
Vectorizer	Stop Words	Basic NLP	Ngram range	Average	Std. Dev.	Average	Std. Dev.	Average	Std. Dev.
count	False	False	(1, 1)	73.66%	1.45%	73.36%	1.59%	76.93%	1.49%
count	False	False	(1, 2)	73.84%	1.70%	71.93%	1.76%	76.17%	1.54%
count	False	False	(1, 3)	73.01%	1.45%	70.84%	1.62%	73.86%	1.48%
count	False	False	(2, 2)	63.60%	1.93%	62.47%	2.05%	67.55%	1.88%
count	False	False	(2, 3)	61.98%	1.81%	61.69%	2.14%	66.66%	1.93%
count	False	True	(1, 1)	73.39%	1.69%	73.12%	1.95%	76.54%	1.66%
count	False	True	(1, 2)	73.32%	1.73%	71.34%	1.68%	75.95%	1.76%
count	False	True	(1, 3)	72.77%	1.76%	70.81%	1.86%	73.60%	1.67%
count	False	True	(2, 2)	63.70%	1.92%	62.65%	2.03%	68.32%	1.90%
count	False	True	(2, 3)	61.75%	1.73%	61.39%	1.76%	66.00%	1.87%
count	True	False	(1, 1)	72.98%	1.92%	73.14%	1.67%	74.05%	1.64%
count	True	False	(1, 2)	72.12%	1.73%	72.60%	1.66%	72.04%	1.59%
count	True	False	(1, 3)	71.57%	1.73%	71.73%	1.71%	69.84%	1.59%
count	True	False	(2, 2)	47.84%	2.05%	50.27%	1.96%	54.09%	1.96%
count	True	False	(2, 3)	44.59%	2.16%	48.36%	1.96%	51.49%	1.85%
count	True	True	(1, 1)	73.37%	1.67%	73.46%	1.71%	74.60%	1.54%
count	True	True	(1, 2)	72.28%	1.75%	72.46%	1.82%	72.73%	1.75%
count	True	True	(1, 3)	71.61%	1.81%	71.60%	1.78%	70.29%	1.77%
count	True	True	(2, 2)	48.09%	1.93%	50.33%	1.89%	54.42%	1.66%
count	True	True	(2, 3)	44.83%	2.12%	48.33%	2.06%	52.15%	1.98%
tfidf	False	False	(1, 1)	76.94%	1.63%	73.41%	1.74%	75.06%	1.62%
tfidf	False	False	(1, 2)	75.23%	1.68%	71.38%	1.62%	74.53%	1.55%
tfidf	False	False	(1, 3)	72.94%	1.56%	70.60%	1.66%	72.09%	1.69%
tfidf	False	False	(2, 2)	64.50%	1.72%	62.09%	2.10%	66.99%	1.89%
tfidf	False	False	(2, 3)	61.05%	2.00%	60.99%	2.13%	65.50%	2.00%
tfidf	False	True	(1, 1)	76.80%	1.52%	73.07%	1.82%	75.17%	1.70%
tfidf	False	True	(1, 2)	74.98%	1.68%	71.12%	1.82%	74.33%	1.53%
tfidf	False	True	(1, 3)	73.00%	1.79%	70.58%	1.75%	72.01%	1.84%
tfidf	False	True	(2, 2)	64.61%	1.97%	61.93%	2.19%	68.53%	1.97%
tfidf	False	True	(2, 3)	61.13%	1.91%	60.79%	1.96%	64.99%	1.71%
tfidf	True	False	(1, 1)	74.67%	1.59%	72.86%	1.55%	71.96%	1.72%
tfidf	True	False	(1, 2)	72.20%	1.68%	72.74%	1.56%	70.12%	1.61%
tfidf	True	False	(1, 3)	70.08%	1.80%	72.04%	1.73%	67.96%	1.76%
tfidf	True	False	(2, 2)	47.58%	1.90%	50.46%	1.93%	54.04%	2.14%
tfidf	True	False	(2, 3)	44.07%	1.82%	48.50%	1.83%	51.36%	1.93%
tfidf	True	True	(1, 1)	74.60%	1.67%	72.89%	1.73%	72.24%	1.73%
tfidf	True	True	(1, 2)	71.89%	1.74%	72.11%	1.73%	70.81%	1.63%
tfidf	True	True	(1, 3)	69.96%	1.85%	71.76%	1.67%	68.25%	2.04%
tfidf	True	True	(2, 2)	47.22%	1.82%	49.89%	1.99%	53.64%	1.91%
tfidf	True	True	(2, 3)	44.33%	2.10%	48.72%	2.19%	52.05%	2.03%

Table 5.1: Classifiers comparison results.